Bases de données

SQL
LMD – Partie 3
Sélections spécialisées

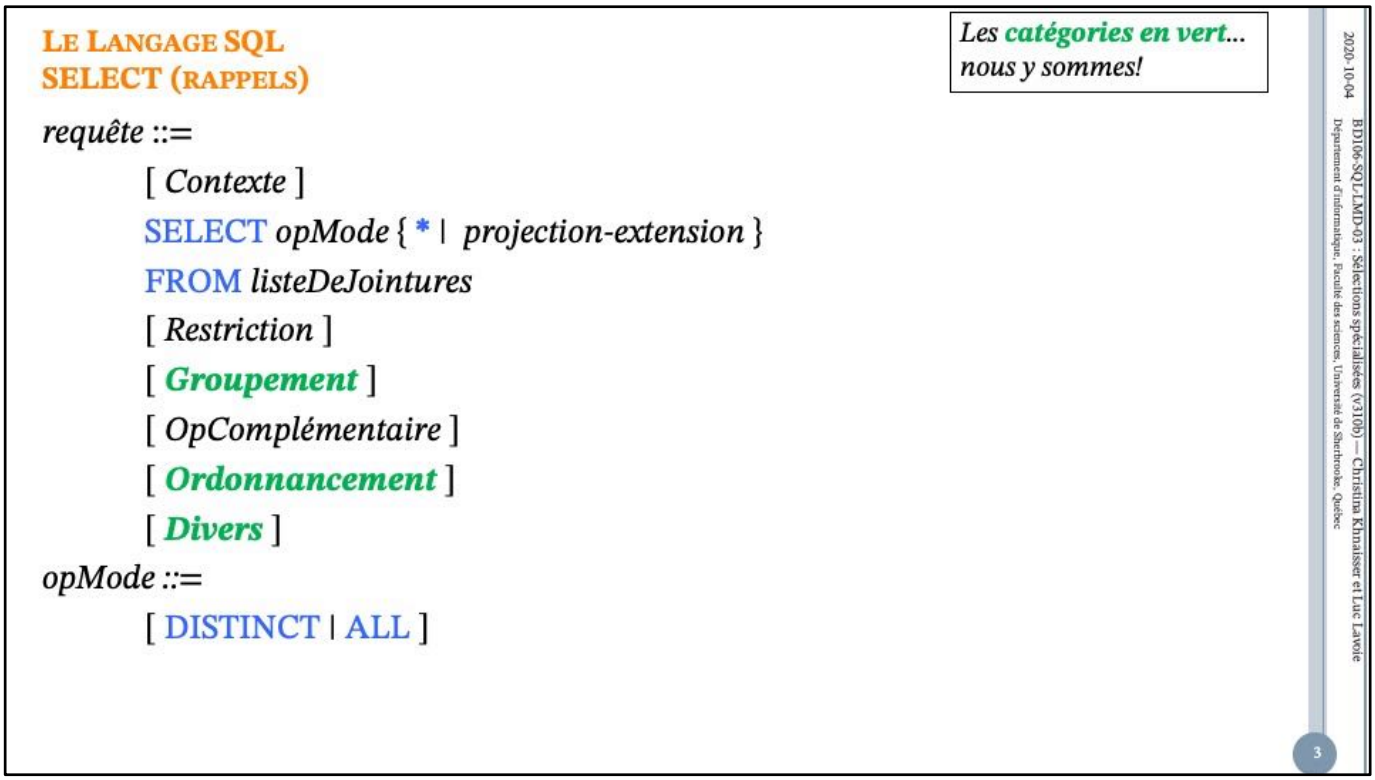
BD106
v310b
2020-10-04

Département d'informatique
Faculté des sciences

UNIVERSITÉ DE SHERBROOKE

Christina.Khnaisser@USherbrooke.ca
http://info.USherbrooke.ca/ckhnaisser
Luc.Lavoie@USherbrooke.ca
http://info.USherbrooke.ca/llavoie

- Rappels
- SELECT
  - Agrégation
  - Groupement
  - Ordonnancement
  - Divers
- Exemples
- Autres variantes



[WITH [RECURSIVE] *requête_with* [, ...]]
SELECT [ALL | DISTINCT [ON (*expression* [, ...])]]
* | *expression* [[AS] *nom_d_affichage*] [, ...]
[FROM *éléments_from* [, ...]]
[WHERE *condition*]
[GROUP BY *expression* [, ...]]
[HAVING *condition* [, ...]]
[WINDOW *nom_window* AS (*définition_window*) [, ...]]
[{ UNION | INTERSECT | EXCEPT } [ALL | DISTINCT] *select*]
[ORDER BY *expression* [ASC | DESC | USING *opérateur*] [NULLS { FIRST | LAST }] [, ...]]
[LIMIT { *nombre* | ALL }]
[OFFSET *début*] [ROW | ROWS]]
[FETCH { FIRST | NEXT } [*total*] { ROW | ROWS } ONLY]
[FOR { UPDATE | SHARE } [OF *nom_table* [, ...]] [NOWAIT] [...]]

avec *éléments_from* qui peut être :

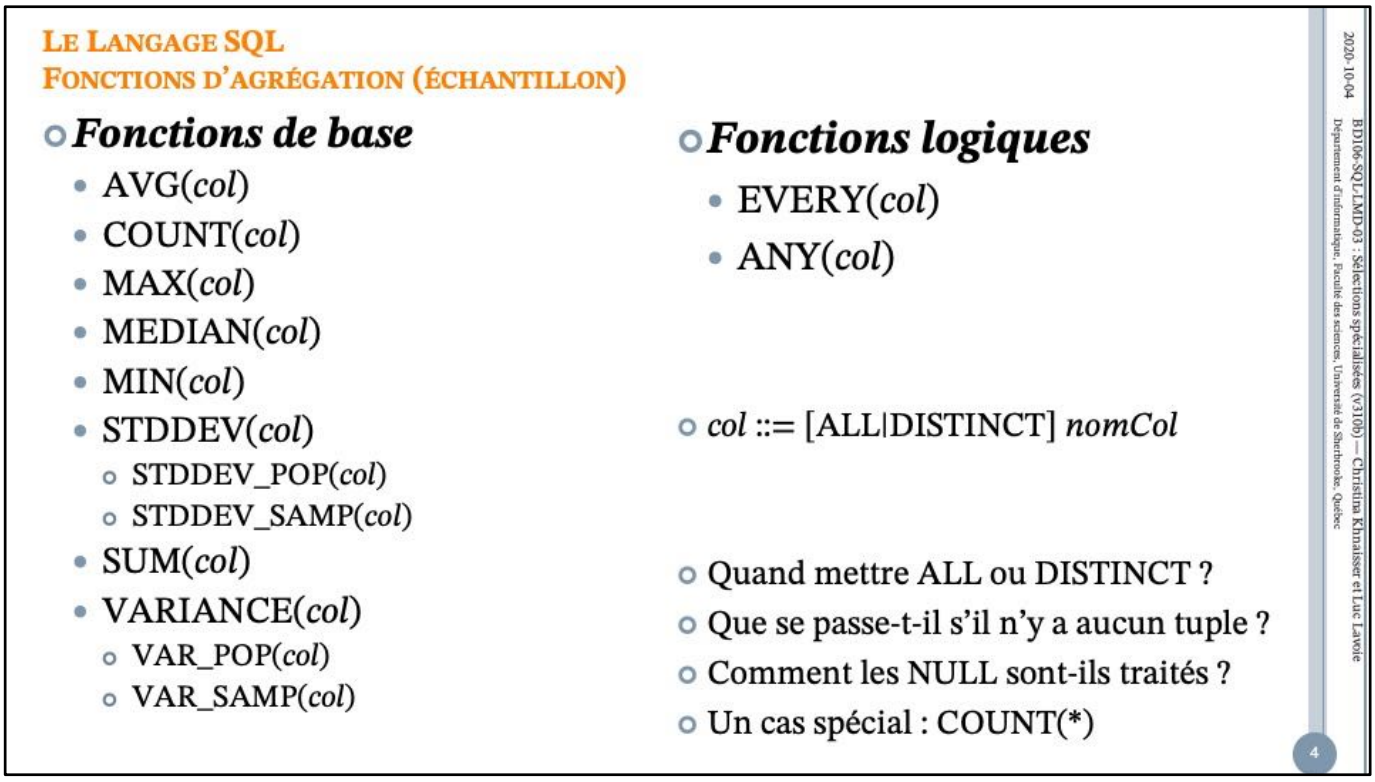
[ONLY] *nom_table* [*] [[AS] *alias* [(*alias_colonne* [, ...])]]
(*select*) [AS] *alias* [(*alias_colonne* [, ...])]
nom_requête_with [[AS] *alias* [(*alias_colonne* [, ...])]]
nom_fonction ([*argument* [, ...]]) [AS] *alias* [(*alias_colonne* [, ...] | *définition_colonne* [, ...])]

nom_fonction ([*argument* [, ...]]) AS (*définition_colonne* [, ...])
éléments_from [NATURAL] *type_jointure éléments_from* [ON *condition_jointure* | USING (*colonne_jointure* [, ...])]

et *requête_with* est :

nom_requête_with [(*nom_colonne* [, ...])] AS (*select* | *valeurs* | *insert* | *update* | *delete*)

TABLE [ONLY] *nom_table* [*]



1) If, during the computation of the result of *AF,* an intermediate result is not representable in the declared type of the site that contains that intermediate result, then

Case:

a) If the most specific type of the result of *AF* is an interval type, then an exception condition is raised:
data exception — interval value out of range.

b) If the most specific type of the result of *AF* is a multiset type, then an exception condition is raised:
data exception — multiset value overflow.

c) Otherwise, an exception condition is raised: *data exception — numeric value out of range.*

2) Case:

a) If <filter clause> is specified, then the <search condition> is applied to each row of *T.* Let *T1* be the collection of rows of *T* for which the result of the <search condition> is *True.*

b) Otherwise, let *T1* be *T.*

3) If COUNT(*) is specified, then the result is the cardinality of *T1.*

4) If <general set function> is specified, then:

a) Let *TX* be the single-column table that is the result of applying the <value expression> to each row of *T1* and eliminating null values. If one or more null values are eliminated, then a completion condition is raised: *warning — null value eliminated in set function.*

b) Case:

i) If DISTINCT is specified, then let *TXA* be the result of eliminating redundant duplicate values from *TX,* using the comparison rules specified in Subclause 8.2, "<comparison predicate>", to identify the redundant duplicate values.

ii) Otherwise, let *TXA* be *TX.*

c) Let *N* be the cardinality of *TXA*

d) Case:

i) If COUNT is specified, then the result is *N.*

ii) If *TXA* is empty, then the result is the null value.

iii) If AVG is specified, then the result is the average of the values in *TXA.*

iv) If MAX or MIN is specified, then the result is respectively the maximum or minimum value in *TXA.* These results are determined using the comparison rules specified in Subclause 8.2, "<comparison predicate>". If *DT* is a user-defined type and the comparison of two values in *TXA* results in *Unknown,* then the maximum or minimum of *TXA* is implementation-dependent.

v) If SUM is specified, then the result is the sum of the values in *TXA.* If the sum is not within the range of the declared type of the result, then an exception condition is raised: *data exception — numeric value out of range.*

vi) If EVERY is specified, then Case:

1) If the value of some element of *TXA* is *False,* then the result is *False.*
2) Otherwise, the result is *True.*

vii) If ANY or SOME is specified, then

Case:

1) If the value of some element of *TXA* is *True,* then the result is *True.*
2) Otherwise, the result is *False.*

viii) If VAR_POP or VAR_SAMP is specified, then let *S1* be the sum of values in the column of *TXA,* and *S2* be the sum of the squares of the values in the column of *TXA.*

1) If VAR_POP is specified, then the result is (S2-S1*S1/N)/N.
2) If VAR_SAMP is specified, then

Case:

A) If *N* is 1 (one), then the result is the null value.

B) Otherwise, the result is (S2-S1*S1/N)/(N-1)

ix) If FUSION is specified, then the result is the multiset *M* such that for each value *V* in the element type of *DT,* including the null value, the number of elements of *M* that are identical to *V* is the sum of the number of identical copies of *V* in the multisets that are the values of the column in each row of *TXA.*

x) If INTERSECTION is specified, then the result is a multiset *M* such that for each value *V* in the element type of *DT,* including the null value, the number of duplicates of *V* in *M* is the minimum of the number of duplicates of *V* in the multisets that are the values of the column in each row of *TXA.*

NOTE 231 — This rule says "the result is a multiset" rather than "the result is the multiset" because the precise duplicate values are not specified. Thus this calculation is non-deterministic for certain element types, namely those based on character

string, datetime with time zone and user-defined types.

ÉVALUATION – EXEMPLE DE DONNÉES (AGG)

Activité

sigle	titre
IFT159	Analyse et programmation
IFT187	Éléments de bases de données
IMN117	Acquisition des médias numériques
IGE401	Gestion de projets
GMQ103	Géopositionnement

TypeÉvaluation

code	description
IN	Examen intra
FI	Examen final
TP	Travail pratique
PR	Projet

Résultat

matricule	TE	activité	trimestre	note
15113150	TP	IFT187	20133	80
15112354	FI	IFT187	20123	78
15113150	TP	IFT159	20133	75
15112354	FI	GMQ103	20123	85
15110132	IN	IMN117	20123	90
15110132	IN	IFT187	20133	45
15112354	FI	IFT159	20123	52

Étudiant

matricule	nom	adresse
15113150	Paul	>ᐃᖏᓗᖅ
15112354	Éliane	Blanc-Sablon
15113870	Mohamed	Tadoussac
15110132	Sergeï	Chandler

2020-10-04 BD106-SQL LMD-03 : Sélections spécialisées (v310b) — Christina Khnaisser et Luc Lavoie
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec
5

Calculer la moyenne des résultats
Calculer la moyenne des résultats du cour IFT187 + min et le max
Calculer la note totale de la matricule 15112354

LE LANGAGE SQL
SELECT (SYNTAXE SIMPLIFIÉE) - GROUPEMENT

Groupement ::=

[GROUP BY { *expression* ... , }]
[HAVING { *condition* ... , }]

- beaucoup de choses à dire ☺
- ...

2020-10-04 BD106-SQLLMD-03 : Sélections spécialisées (v310b) — Christina Khnaisser et Luc Lavoie
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec
6

ÉVALUATION – EXEMPLE DE DONNÉES (GROUP)

Résultat

matricule	TE	activité	trimestre	note
15113150	TP	IFT187	20133	80
15112354	FI	IFT187	20123	78
15113150	TP	IFT159	20133	75
15112354	FI	GMQ103	20123	85
15110132	IN	IMN117	20123	90
15110132	IN	IFT187	20133	45
15112354	FI	IFT159	20123	52
15112354	TP	IFT159	20123	50

Étudiant

matricule	nom	adresse
15113150	Paul	>ᐃᖅᓂᑐᖅ
15112354	Éliane	Blanc-Sablon
15113870	Mohamed	Tadoussac
15110132	Sergeï	Chandler

2020-10-04 BD106-SQLLMD-03 : Sélections spécialisées (v310b) — Christina Khnaisser et Luc Lavoie
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec
7

Le nombre d'activités auxquelles un étudiant est ou a été inscrit :

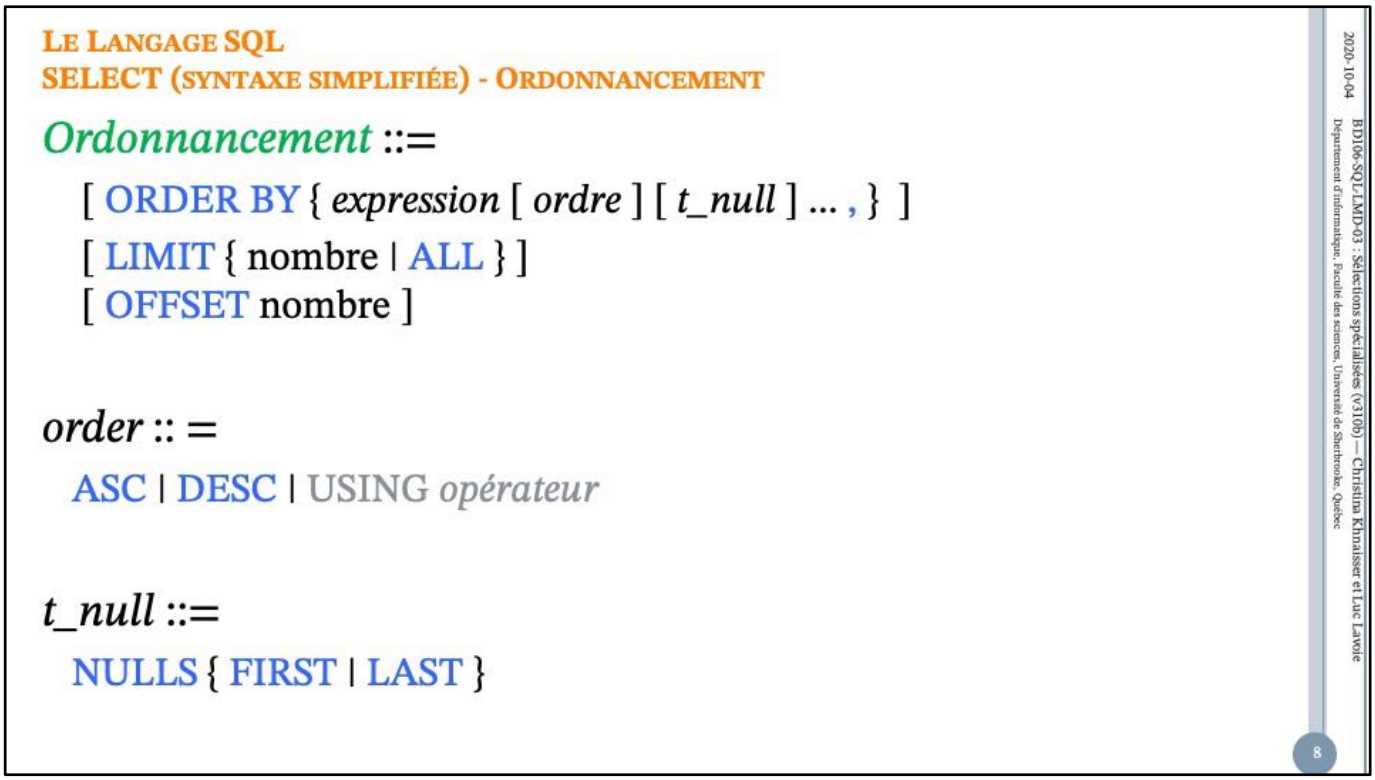
```
SELECT matricule, COUNT(activite) AS nbActivites
FROM resultat
GROUP BY matricule
```

Combien de fois par année une activité pédagogique est elle donnée ?

```
SELECT activite, COUNT(DISTINCT trimestre) AS nbFois
FROM resultat
GROUP BY activite
```

Note totale par étudiant pour chacun de ces cours

```
SELECT matricule, activite, SUM(note) AS noteTotale
FROM resultat
GROUP BY matricule, activite
```



[ORDER BY *expression* [ASC | DESC | USING *opérateur*] [NULLS { FIRST | LAST }] [, ...]]
[LIMIT { *nombre* | ALL }]
[OFFSET *nombre*] [ROW | ROWS]]
[FETCH { FIRST | NEXT } [*total*] { ROW | ROWS } ONLY]
[FOR { UPDATE | SHARE } [OF *nom_table* [, ...]] [NOWAIT] [...]]

LIMIT n : les n premiers tuples sont extraits selon l'ordre spécifié
OFFSET n : tous les tuples sont extraits sauf les n premiers selon l'ordre spécifié (le complément de LIMIT, si exclusif)

Qu'arrive-t-il si les deux sont présents ?
Sont-ils appliqués dans l'ordre ou l'un a-t-il préséance sur l'autre ?

ÉVALUATION – EXEMPLE DE DONNÉES (ORDER)

Activité

sigle	titre
IFT159	Analyse et programmation
IFT187	Éléments de bases de données
IMN117	Acquisition des médias numériques
IGE401	Gestion de projets
GMQ103	Géopositionnement

TypeÉvaluation

code	description
IN	Examen intra
FI	Examen final
TP	Travail pratique
PR	Projet

Résultat

matricule	TE	activité	trimestre	note
15113150	TP	IFT187	20133	80
15112354	FI	IFT187	20123	78
15113150	TP	IFT159	20133	75
15112354	FI	GMQ103	20123	85
15110132	IN	IMN117	20123	90
15110132	IN	IFT187	20133	45
15112354	FI	IFT159	20123	52

Étudiant

matricule	nom	adresse
15113150	Paul	>ᐃᖏᓂᑲᖅ
15112354	Éliane	Blanc-Sablon
15113870	Mohamed	Tadoussac
15110132	Sergeï	Chandler

2020-10-04 BD106-SQLLMD-03 : Sélections spécialisées (v310b) — Christina Khnaisser et Luc Lavoie
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

9

Extraire la note la plus haute : ORDER BY note DESC Limit 1 ou ORDER BY note ASC Offset 1
Extraire la note la plus basse : ORDER BY note DESC Offset 1

Divers ::=

FETCH { FIRST | NEXT } [total] { ROW | ROWS } ONLY]
[FOR { UPDATE | SHARE }]
| WINDOW ...
| RANK ...

... et bien d'autres choses encore

Exercices

- Le répertoire public de cours
  - smb://dinf-argus.dinf.fsci.usherbrooke.ca/public/Cours/IFT187/Exemples
- propose plusieurs exemples dont les suivants
  - Evaluation
  - Pharmacie
  - Gaspard

RÉFÉRENCES

- Elmasri et Navathe (4e ed.), chapitre 7
- Elmasri et Navathe (6e ed.), chapitre 4
- [Loney2008]
  Loney, Kevin ;
  Oracle Database 11g: The Complete Reference.
  Oracle Press/McGraw-Hill/Osborne, 2008.
  ISBN 978-0071598750.
- [Date2012]
  Date, Chris J. ;
  SQL and Relational Theory : How to Write Accurate SQL Code.
  2nd edition, O'Reilly, 2012.
  ISBN 978-1-449-31640-2.
- Le site d'Oracle (en anglais)
  - http://docs.oracle.com/cd/E11882_01/index.htm
- Le site de PostgreSQL (en français)
  - http://docs.postgresqlfr.org